

WE CLAIM:

1. A processor implemented method of generating a build for a software product from one or more source-files, the method comprising:

5 i) processing each of the one or more source-files to remove comment lines from said source file to produce a respective compacted source file; and

ii) compiling each of the one or more compacted source-files to generate a new build of the software product.

10 2. The processor implemented method according to claim 1 further comprising, for each of the one or more source files, a step of determining whether or not it is sufficiently beneficial to remove comment lines from said source file, and if it is not sufficiently beneficial not removing the comment
15 lines from said source file in a subsequent iteration of the method.

3. The processor implemented method according to claim 1 adapted for use with C/C++.

4. The processor implemented method according to claim 1
20 adapted for use with a programming language which supports modular program construction through compile time processing of the one or more source-files.

5. The processor implemented method according to claim 2 further comprising a step of counting comment lines in each of
25 the one or more source files and the number of times each of the one or more source files is included in another of the one or more source-files, wherein the step of determining whether or not it is sufficiently beneficial to remove comment lines from a particular source file is based on the number of comment
30 lines in the particular source file and the number of times

that particular source file is included in another of the one or more source files.

6. A method according of claim 1 further comprising between the generation of successive builds, adaptively
5 selecting which source-files to be processed for comment removal.

7. The method according to claim 2 further comprising for each of the one or more source-files generating comment expansion statistics, wherein the step of determining whether
10 or not it is sufficiently beneficial to remove comment lines from a particular source file is based on the comment expansion statistics for that particular source file.

8. The method according to claim 7 wherein the comment expansion statistics are generated periodically for each of the
15 one or more source files.

9. An apparatus for generating a build for a software product from one or more source-files, the apparatus comprising:

i) a comment extraction program adapted to process
20 each of the one or more source-files to remove comment lines from said source file to produce a respective compacted source file; and

ii) a compiler adapted to compile each of the one or more compacted source-files to generate a new build of the
25 software product.

10. The apparatus of claim 9 further comprising a code repository in which the one or more source-files and the respective compacted source files are stored.

11. The apparatus of claim 9 further comprising a pre-processor for pre-processing each of the one or more compacted source-files before the compiler compiles.

12. The apparatus of claim 9 further comprising a pre-processor for pre-processing each of the one or more source files before the comment extraction program removes the comment lines to produce the respective compacted source-file.

13. The apparatus of claim 10 further comprising an extraction program adapted to remove each of the one or more source-files from the code repository.

14. The apparatus of claim 13, wherein the extraction program and the comment extraction program are integrated with one another so that as each of the one or more source-files is removed from the code repository comment lines in each of the one or more source-files are removed.

15. A computer readable medium having instructions stored thereon for implementing a method of generating a build for a software product made-up of one or more source-files, the method comprising:

20 i) processing each of the one or more source-files to remove comment lines from said source file to produce a respective compacted source file; and

ii) compiling each of the one or more compacted source-files to generate a new build of the software product.

25 16. The computer readable medium of claim 15, wherein the method further comprises for each of the one or more source files determining whether or not it is sufficiently beneficial to remove comment lines from said source file, and if it is not sufficiently beneficial not removing the comment lines from
30 said source file in a subsequent iteration of the method.

17. The computer readable medium of claim 16, wherein the method further comprises counting comment lines within each of the one or more source files and the number of times each of the one or more source files is included in another of the one or more source-files, wherein determining whether or not it is sufficiently beneficial to remove comment lines from a particular source file is based on the number of comment lines in the particular source file and the number of times that particular source file is included in another of the one or more source files.

18. The computer readable medium of claim 15, wherein the method further comprises, between the generation of successive builds, adaptively selecting which source-files to be processed for comment line removal.

19. The computer readable medium of claim 16, wherein the method further comprises for each of the one or more source-files generating comment expansion statistics, wherein determining whether or not it is sufficiently beneficial to remove comment lines from a particular source file is based on the comment expansion statistics for that particular source file.

20. The computer readable medium of claim 19, wherein for the method the comment expansion statistics are generated periodically for each of the one or more source files.